Compact Speech Translation Models via Discrete Speech Units Pretraining

Tsz Kin Lam and **Alexandra Birch** and **Barry Haddow**
School of Informatics, University of Edinburgh
{tlam, a.birch, bhaddow}@ed.ac.uk

Abstract

We propose a pretraining method to use Self-Supervised Speech (SSS) model to creating more compact Speech-to-text Translation. In contrast to using the SSS model for initialization, our method is more suitable to memory constrained scenario such as on-device deployment. Our method is based on Discrete Speech Units (DSU) extracted from the SSS model. In the first step, our method pretrains two smaller encoder-decoder models on 1) Filterbank-to-DSU (Fbk-to-DSU) and 2) DSU-to-Translation (DSU-to-Trl) data respectively. The DSU thus become the distillation inputs of the smaller models. Subsequently, the encoder from the Fbk-to-DSU model and the decoder from the DSU-to-Trl model are taken to initialise the compact model. Finally, the compact model is finetuned on the paired Fbk-Trl data. In addition to being compact, our method requires no transcripts, making it applicable to low-resource settings. It also avoids speech discretization in inference and is more robust to the DSU tokenization. Evaluation on CoVoST-2 (X-En) shows that our method has consistent improvement over the baseline in three metrics while being compact i.e., only half the SSS model size.

1 Introduction

In Speech-to-text Translation (ST), using Self-Supervised Speech (SSS) models, such as wav2vec 2.0 and HuBERT (Baevski et al., 2020; Hsu et al., 2021), as model initialization is now common to obtain the SOTA result (Agarwal et al., 2023). Nevertheless, such model initialisation makes the ST model less memory-adaptive and could impose a large memory footprint. These factors hinders on-device deployment that is crucial for privacy and useful in the absence of internet connection.

How can we use the SSS model(s) to create a more compact ST model? When using the SSS model for initialization, the corresponding ST model uses the dense representations of the SSS model for its task. Alternatively, an informative proxy, which requires less memory to obtain, for the dense representation may make the ST model more compact.

Discrete Speech Units (DSU) extracted from the SSS model can be such a good proxy. DSU are K-Means clusters of speech representations from selected layers of the SSS model. It represents sequence of discrete tokens, which are easier to model within a text processing architecture (Polyak et al., 2021; Chou et al., 2023). DSU sequences[1] are far smaller than the sequences of dense representations. Therefore, a straightforward method to distill the SSS models is to use DSU as speech inputs, aka the DSU-to-Translation (DSU-to-Trl) model. Although using DSU as inputs allows for transfer learning and a memory-adaptive model, using them at inference still requires storing and calling the quantization modules, i.e, the SSS model and the K-Means model.

We thus propose to use DSU for pretraining (PT) rather than as model input to make ST models more compact. Our method distils the SSS model by pretraining smaller models on the corresponding DSU. More specifically, our method firstly pretrains two smaller encoder-decoder models on 1) Filterbank-to-DSU (Fbk-to-DSU) and 2) DSU-to-Trl data respectively. The DSU thus become the distillation inputs of the smaller models. Subsequently, the encoder from the Fbk-to-DSU model and the decoder from the DSU-to-Trl model are taken to initialise the compact model. Finally, the compact model is finetuned on the paired Fbk-Trl data. Under this formulation, (1) we can use the SSS model to create a ST model that is adaptive to the memory footprint. (2) Our method requires no transcripts, unlike ASR-pretraining, making it applicable to low-resource

[1] In this paper, DSU and DSU sequences are used interchangeably. When we need to focus on a few units of the sequence, we call them DSU tokens.

Proceedings of the 21st International Conference on Spoken Language Translation (IWSLT 2024), pages 162–172
August 15-16, 2024

settings. (3) Our method avoids using the quantization modules in inference. (4) Extensive results also show that our method is more robust to DSU tokenization than the DSU-to-Trl method.

We evaluate our method on CoVoST-2 (Wang et al., 2021) X-En language directions (21 in total) using multilingual ST. By using a HuBERT-Base model to extract the DSU, our method shows strong and consistent improvements in three evaluation metrics with respect to a ST model that is trained from scratch. Our main contributions are:

- We propose a pretraining method to distil the SSS model to creating a more compact ST model. Rather than competing with the SOTA ST models, adaptability to the memory footprint is our key focus.
- Our method uses DSU for pretraining rather than as model inputs. This lowers the inference cost, especially for on-device purpose, by avoiding the quantization modules (storage and running).
- We conduct extensive analysis to study the effect of DSU tokenization to both using DSU as model inputs and as pretraining. Our pretraining method is found to be more robust to different tokenizations.

2 Related Work

There are a number of related works that use DSU to enhance ST. Fang and Feng (2023) and Zhang et al. (2023b) use DSU to create more training data in a back-translation fashion. Chang et al. (2023) and Zhang et al. (2023b) explore the replacement of Filterbank by DSU as speech input. Furthermore, Yan et al. (2024) proposes a multi-tasking learning framework with hard parameter sharing, i.e., using a joint vocabulary for text tokens and DSU, to improve the speech-text modality gap. In contrast, we use DSU and its translation model for pretraining, resulting in a better Fbk-to-Trl model that has a shorter inference pipeline.

In the case of pretraining, Wu et al. (2023) use a single Speech-to-DSU model in pretraining for general speech-to-text purposes whereas we tailor the use for ST by using a pair of encoder-decoder models. Zhang et al. (2022b) also decompose ST into speech-to-unit and unit-to-text tasks. Their training is based on masked unit prediction, and it requires an extra unit-encoder module in inference. In contrast, we resort to supervised training on the DSU in acoustic pretraining and require no extra module in inference. More importantly, our goal is to make (multilingual) ST more compact, aiming also at low-resource settings where transcripts are not easily available, rather than learning a joint semantic space for both transcripts and audios.

3 Method

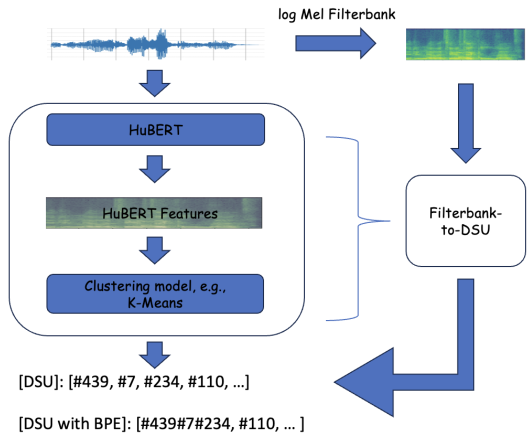

Figure 1: Illustration of the Fbk-to-DSU model. It is like an auto-encoding training process, but between a continuous format (log Mel Filterbank) and its discrete format (DSU) that is extracted from a HuBERT model.

Our method uses DSU in the form of pretraining to distil knowledge from the SSS (dense) representations to creating a more compact ST model.

In the first step, our method pretrains two smaller encoder-decoder models on 1) Fbk-to-DSU and 2) DSU-to-Trl data respectively. The Fbk-to-DSU model takes the log Mel Fbk as the encoder input and predicts the DSU sequence. The model is trained by an interpolation of Connectionist Temporal Classification (CTC, Graves et al. (2006)) loss that is applied to the last encoder layer and label-smoothed Cross-Entropy (CE) loss:

$$\mathcal{L}^{\text{Fbk-to-DSU}} = (1-\lambda_\alpha)\mathcal{L}_{\text{CE}}(\mathbf{U}|\mathbf{F})+\lambda_\alpha\mathcal{L}_{\text{CTC}}(\tilde{\mathbf{U}}|\mathbf{F}) \quad (1)$$

where $\mathbf{F} \in \mathbb{R}^{T\text{x}D}$, $\mathbf{U} \in \mathcal{U}$ and $\tilde{\mathbf{U}} \in \tilde{\mathcal{U}} = \{\mathcal{U}, \mathit{blank}\}$ are the Fbk, DSU and the CTC label sequences respectively. The CTC vocabulary correspond to an union of the same vocabulary used in the CE loss and a *blank* label. The idea is similar to an autoencoder, but the Fbk-to-DSU model is trained to map the Fbk inputs to its discrete form from the SSS model in a multi-task learning fashion (Figure 1). The DSU-to-Trl model learns via CE to predict the translations $\mathbf{Y}$ given $\mathbf{U}$: $\mathcal{L}^{\text{DSU-to-Trl}} = \mathcal{L}_{\text{CE}}(\mathbf{Y}|\mathbf{U})$. In essence, we use the DSU to bridge the speech and text modalities.

Next, we use the encoder of the Fbk-to-DSU model and the decoder (and its output layer) of the DSU-to-Trl model to initialise the compact model, followed by finetuning on the paired Fbk-Trl data using both CE and CTC loss (Gaido et al., 2021; Zhang et al., 2023a) on the translations:

$$\mathcal{L}^{\text{FT}} = (1 - \lambda_\beta)\mathcal{L}_{\text{CE}}(\mathbf{Y}|\mathbf{F}) + \lambda_\beta \mathcal{L}_{\text{CTC}}(\tilde{\mathbf{Y}}|\mathbf{F}) \quad (2)$$

where $\tilde{\mathbf{Y}} \in \tilde{\mathcal{Y}} = \{\mathcal{Y}, blank\}$.

3.1 Tokenization of DSU in different models

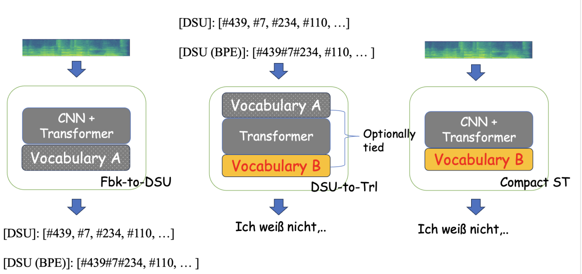

Figure 2: Aligning the DSU tokenization of the Fbk-to-DSU, DSU-to-Trl and compact ST model.

The discrete nature of DSU makes the above training process similar to the transcripts-based pretraining. However, DSU is self-supervised, whereas transcripts require human annotations. DSU are also much longer and can be represented with various sets of symbols.

The length issue could be relieved by merging sequential repetitions (Ao et al., 2022), e.g., '#1 #1 #1 #456 #456 #23' becomes '#1 #456 #23', where each DSU token is denoted by a #{integer}. Byte Pair Encoding (BPE) (Sennrich et al., 2016) could be applied to reduce the DSU sequence length further, e.g., '#1 #456 #23' could be split into a single subword unit: '#1#456#23'.

Since both Fbk-to-DSU and DSU-to-Trl models map to different targets, and DSU can be represented with various set of symbols, we align the tokenizations (or called vocabularies[2]) of the two models. Figure 2 provides an illustration. The vocabulary of the Fbk-to-DSU model (Vocabulary A) is identical to the source vocabulary of the DSU-to-Trl model (their weights are not shared since these two models are trained independently), whereas the target vocabulary (Vocabulary B) of the DSU-to-Trl model is identical to the target vocabulary of the final compact model (their weights are shared during initialisation). The DSU-to-Trl model is similar to a text translation model, so we also experiment of using separate vocabularies or a joint vocabulary. If a joint vocabulary of English subword units and DSU (BPE or not) is used, all the three models would have the same vocabulary, and the weights of the source and target vocabularies of the DSU-to-Trl model are also tied.

4 Experiments

4.1 Data Preprocessing

We follow standard practices to preprocess the CoVoST-2 X-En data. For speech inputs using 80-D log Mel Fbk, we computed the features for every 10ms with a 25ms window and then normalized them using its mean and variance computed over each channel. We use the BPE implementation from SENTENCEPIECE (Kudo and Richardson, 2018) and obtain vocabulary of size 8K on the English target, 16K on the (non-English) transcripts and 32K on the DSU, unless otherwise specified.

We use HuBERT-Base[3] model to extract the DSU by first downsampling the CoVoST-2 audio to 16KHz. Each audio data utterance is then converted into the DSU, i.e., the clustering indexes, by applying K-Means clustering (K=1,000; `MiniBatchKMeans` from SKLEARN) on its HuBERT representation from the 6th layer (Lakhotia et al., 2021). To train the K-Means model, we divide the 21 language pairs into three groups: 1) {ar, cy, et, id, ja, lv, mn, sl, sv, ta, tr}, 2) {nl, pt, ru, zh} and 3) {ca, de, es, fa, fr, it}. We then sample 1K instances for each language pair in group 1), which becomes 3K in group 2) and 12.5K in group 3), to create a multilingual training dataset of 98K instances for the K-Means model.

4.1.1 On the choice of using HuBERT-Base

Given the rapid advance in the SSS models, there are many alternatives, such as XLS-R (Babu et al., 2021) and Wavlm (Chen et al., 2022), for extracting the DSU for our method. These models are larger in scale and could be multilingual, thus providing DSU of higher qualities. The improvement of our method by using DSU from the HuBERT-Base would probably be a lower-bound, considering its

[2]We use vocabulary and tokenization interchangeably, since we did not apply subword regularisation.

[3]https://github.com/facebookresearch/fairseq/tree/main/examples/textless_nlp/gslm/speech2unit

relatively poor qualities to the bigger models. Since our goal is about compactness via DSU pretraining rather than comparing the DSU qualities across the SSS models, we took a simple HuBERT-Base model to illustrate the idea. Pretraining only on English audio data could also suggest hints on whether the DSU and our method could be generalised to languages that are unseen to the SSS models.

4.2 Model Configuration

All models are based on Transformer (Vaswani et al., 2017) with implementations from FAIRSEQ (Ott et al., 2019; Wang et al., 2020). In the Fbk-to-Token (i.e. transcriptions, DSU, or translations) models, the encoder has convolutional layers to downsample the Fbk by a factor of 4. There are 12-6 layers in the transformer encoder-decoder, whereas the embedding and feed-forward network (FFN) dimensions are 256 and $4,096$ respectively, unless otherwise specified. It is worth noting that: (1) The Fbk-to-DSU model is not trained on the translations, so it is not directly comparable to the ST models. Its effect on ST lies on its pretrained encoder (Table 2). (2) The DSU-to-Trl model is a ST model which decoder can be used for initialization.

Scratch is a ST model trained on the paired speech-translation data without pretraining.

ASR Pretraining refers to a ST model whose encoder is initialized by a speech recognition task with CTC regularisation on the transcripts.

DSU-to-Trl follows the Transformer used in text translation. We use 6-6 layers in the encoder-decoder which the dimension of embedding and FFN is 256 and $2,048$ respectively. In addition, we use "pre" layer-normalization (Nguyen and Salazar, 2019). Despite its smaller model size, its inference requires the quantization modules.

Hu-Transformer uses the entire HuBERT as the speech encoder initialization (Fang and Feng, 2023). For comparison to our DSU-Adapter, its subsequent encoder-decoder also has 1-6 layers.

DSU-Adapter is our proposed method. To better align the two pre-trained components, we also experiment with adding an extra encoder layer as a simple adapter layer after the pre-trained encoder. Because of the small model size, all model parameters are trainable. Since its decoder is initialized by the DSU-to-Trl method, its decoder FFN dimension is $2,048$.

Enc-Init is a ST model that has its encoder initialized by the Fbk-to-DSU encoder. **EncDec-Init** is a DSU-Adapter model without the adapter layer.

4.3 Training and Inference

It is worth noting that we do not use extra audio data, e.g., Libri-Light (Kahn et al., 2020) in our (pretraining) experiments. Furthermore, we apply the following conditions in (pre-)training:

- We skip training data that are longer than 30 seconds (audio) or $1,024$ target tokens.
- We apply SpecAugment (Park et al., 2019) with parameters: $\{F = 30, T = 40, m_F = 2, m_T = 2\}$ on Filterbank inputs.
- We share the embedding weights when using a joint vocabulary in the DSU-to-Trl model.
- We set λ_α and λ_β in CTC to 0.3 and the smoothing parameter to 0.1
- We initialize the encoder (decoder) with the last (best) checkpoint from the PT model.
- We use Adam optimizer with inverse square root scheduler for all model training.
- In all Fbk-to-Token models, the *effective mini-batch size*, *warm-up steps*, *peak learning rate* and *training steps* are 32K frames, 25K, 2e$-$3 and 60K steps respectively.
- Similarly, in all DSU-to-Trl models, we use 80K tokens, 10K, 5e$-$4 and 50K steps.
- Similarly, in Hu-Transformer, we use 4M frames, 4K, 1e$-$4 and 300K steps.

In inference, we average the last 5 checkpoints and use beam size of 5 in generation. All experiments are run on Nvidia A100 GPUs. It takes about 1 day for 2 A100 (40GB) GPUs to complete an experiment that uses Filterbank as speech inputs.

5 Results and Analysis

Before discussing the results, it is worth noting that (1) *Hu-Transformer is not memory-adaptive*, and (2) *ASR-Pretraining requires transcripts, unlike DSU which is self-supervised*. Both methods are introduced for reference purposes of if such resources are available.

AST model (#Params)	BLEU				chrF				COMET-22-DA			
	High	Mid	Low	All	High	Mid	Low	All	High	Mid	Low	All
Scratch (52M)	19.4	7.91	0.73	5.99	43.6	27.2	14.6	23.1	0.605	0.498	0.433	0.481
ASR-Pretraining (52M)	26.5	12.2	1.82	9.00	51.9	32.8	16.4	27.1	0.680	0.537	<u>0.443</u>	0.511
Hu-Transformer (113M)	24.3	11.4	<u>2.18</u>	8.60	49.9	31.9	<u>17.0</u>	26.8	0.650	0.522	0.439	0.499
DSU-Adapter (48M)	<u>26.5</u>	<u>12.9</u>	1.76	<u>9.13</u>	<u>52.1</u>	<u>33.9</u>	16.5	<u>27.4</u>	<u>0.681</u>	<u>0.548</u>	0.442	<u>0.513</u>

Table 1: Results in BLEU, chrF and COMET-22-DA on the test set of CoVoST-2 (X-En) by resource group. In all metrics, DSU-Adapter is much better than Hu-Transformer, which is 2.3 times larger, in both "High" and "Mid" groups. DSU-Adapter, which does not requires transcripts in training, is also on a par with ASR-Pretraining. The best result in each group is denoted by '_'.

5.1 Improvement brought by DSU-Adapter

We divide the 21 language pairs by resource level into: 1) "High": {ca, de, es, fr}, 2) "Mid": {fa, it, pt, ru and zh}, 3) "Low": {ar, cy, et, id, ja, lv, mn, nl, sl, sv, ta, tr} and 4) "All": the 21 languages pairs. We report the average BLEU[4] and chrF[5] over the test sets of each group using SACREBLEU (Post, 2018). In addition, we also provide the result in WMT22-COMET-DA (Rei et al., 2022), which the source inputs are the gold-reference transcripts.

Table 1 compares our DSU-Adapter and the baselines. Our DSU-Adapter is 3 BLEU (in the group "All") higher than the Scratch model. This shows that our proposed method of using DSU-pretraining can strengthen direct end-to-end ST without requiring transcripts and remain flexible in memory footprint (smaller in size than the HuBERT model). Furthermore, it is better than Hu-Transformer in spite of having half the parameters. For "Mid" and "High", the improvement in BLEU is 1.49 and 2.23 points respectively, but it falls short by 0.42 points for "Low". We also compare to ASR pretraining, which is not always applicable, e.g., in low-resource setting or perhaps even in an unwritten language (Zhang et al., 2022a). Surprisingly, our adapter is on a par with it, and its BLEU is 0.13 points better. The result remains consistent when it is measured in chrF and COMET.

5.1.1 Language-specific performance

Figure 3 shows the performance on each language pair in BLEU, chrF and COMET-22-DA. Our DSU-Adapter (in green triangles) show consistent improvement over the Scratch model (in blue circles) in all language pairs. Such improvement is rather surprising since HuBERT-Base was trained solely on English audio data. We hypothesized that the cross-lingual improvement is related to HuBERT's ability to capture language independent features, e.g. phonetic properties (Pasad et al., 2023).

Compared with Hu-Transformer, DSU-Adapter maintains an evident improvement over most language pairs in both "High" and "Mid" groups. Exceptions are in "fa" and "pt", but the lags are almost negligible. In group "Low", Hu-Transformer is slightly better, especially in "nl" and "sv" pairs. However, most translation in this group is barely around 2 BLEU, and the lags are small.

In most language pairs, DSU-Adapter performs similarly to ASR-pretraining (in red diamonds), except translating from "ru" audios. The improvement in this "ru-en" pair makes DSU-Adapter to have an evident advantage of 0.7 BLEU in the group "Mid".

5.2 Tokenization effect to the DSU-to-Trl method and the DSU-Adapter method

In this section, we investigate how tokenization, including BPE, affects the DSU-to-Trl method and the DSU-Adapter method. We are particularly interested in their robustness toward the tokenization, especially using BPE on the DSU, since tuning the quantization process and retraining the subsequent models is computationally expensive.

In Table 2, the 1st column "Has BPE on DSU?" indicates if BPE is applied on the DSU. If "Yes", multiple DSU could be merged into one subword unit, e.g., '#1 #456 #23 #999' could be merged into '#1#456#23#999'. The 2nd column "$|\mathcal{V}|$" shows the vocabulary configuration: its size, and if the model has a joint vocabulary. For example, "1K-8K" means that we use a vocabulary of size 1K for DSU and a second vocabulary of size 8K for English so that the DSU-to-Trl model would have separate vocabularies for the source (DSU) and target (English) sides. All results are in BLEU averaged over all language pairs, i.e., group "All".

[4] nrefs:1|case:mixed|eff:no|tok:13a|smooth:exp|version:2.3.1
[5] nrefs:1|case:mixed|eff:yes|nc:6|nw:0|space:no|version:2.3.1

Has BPE on DSU?	$\lvert\mathcal{V}\rvert$	DSU Length	Length Ratio	DSU-to-Trl (20M to 27M)	Enc-Init (52M to 70M)	EncDec-Init (46M to 64M)	DSU-Adapter (48M to 67M)
No	1K-8K	176	12.9	6.73	7.70	7.87	8.54
	1K-16K	"	14.1	6.36	7.50	8.00	8.43
	1K-32K	"	14.9	6.30	7.23	7.65	7.94
	8K	"	12.7	6.88	7.64	8.05	8.26
	16K	"	14.0	6.33	7.41	7.91	8.17
	32K	"	14.9	6.26	6.66	7.41	7.68
Yes	1K-8K	221	16.3	4.52	8.23	8.44	8.61
	16K-8K	129	9.5	5.06	8.51	8.76	8.95
	32K-8K	115	8.5	4.43	8.67	9.02	9.13
	8K	150	7.6	7.02	8.33	8.51	8.82
	16K	133	7.7	6.50	8.57	8.61	8.93
	32K	118	7.8	5.07	8.30	8.44	8.70

Table 2: (DSU) tokenization effect on 4 ST methods. Each ST model's performance on the CoVoST-2 test set is measured by BLEU on group "All". All 4 methods could perform better than the Scratch model of 5.99 BLEU as shown on Table 1. In general, darker (brighter) cells refer to weaker (stronger) models. The best two models apply both BPE on the DSU and separate vocabularies in PT (cells in yellow).

5.2.1 DSU-to-Trl: robust to tokenization?

When BPE is not applied on the DSU, those 6 DSU-to-Trl models have 6.48 ± 0.26 BLEU. Despite having smaller model size (<30M), they are better than the Scratch model of 5.99 BLEU.

When BPE is applied, the sequence length of DSU (DSU Length) could be shortened, which could in turn improve the performance, e.g. the best DSU-to-Trl model happens at configuration "8K" with 7.02 BLEU. However, the DSU-to-Trl method is quite unstable to the use of BPE, as reflected by the 5.12 ± 0.83 BLEU in the other 5 configurations. The correlation between the DSU sequence length, the source-target length ratio, and the ST performance is also not straightforward. For an example, the "32K" model (DSU length of 118) is about 2 BLEU behind to the "8K" model (DSU length of 150). Therefore, applying BPE on the DSU for length reduction should remain cautious.

5.2.2 The DSU-Adapter is more robust

Unlike DSU-to-Trl method, DSU-Adapter benefits more when BPE is applied to the DSU. Our proposed method has 8.86 ± 0.19 BLEU (over the 6 corresponding configurations), as opposed to 8.17 $\pm$ 0.32 BLEU when BPE is not applied. This observation is opposed to the DSU-to-Trl method which only scores 5.54 ± 1.07 BLEU (with also larger variance) when BPE is applied on the DSU but 6.48 ± 0.26 when BPE is not used. The improved mean score and its smaller variance suggests that the DSU-Adapter method is more (DSU) tokenization robust. We see this as a benefit of introducing the DSU, i.e., the SSS model knowledge, via PT rather than as model inputs.

On top of applying BPE on the DSU, using separate vocabularies in PT is preferred (the two yellow cells on Table 2) since it performs slightly better, and the DSU, which are not needed in the ST output, would not occupy the target vocabulary.

5.2.3 Ablation: initialisation in DSU-Adapter

Having similar model sizes, e.g. about 50M parameters (Table 2), DSU-Adapter is better than both EncDec-Init and Enc-Init methods. The translation performance in BLEU (averaged over the 12 vocabularies) is 8.51 ± 0.44, 8.22 ± 0.51, and 7.99 ± 0.61 respectively. Encoder-initialization seems more crucial than decoder-initialization, as reflected by the fact that the best DSU-Adapter model comes from a combination with the weakest DSU-to-Trl model of 4.43 BLEU.

5.3 Is CTC applicable also to DSU?

Similar to ST methods that use pretrained components, our method could be limited by the *pretraining modality gap* (Liu et al., 2020; Le et al., 2023). Motivated by prior works, we investigate mitigating it with CTC. A crucial difference to the prior works is that our method uses DSU for pre-training rather than transcripts.

We thus study applying CTC in our method at

Has CTC in DSU PT?	Has CTC in ST FT?	High	Mid	Low	All
No	No	25.81	9.91	1.46	8.14
No	Yes	26.10	11.35	1.69	8.71
Yes	No	25.94	10.82	1.51	8.44
Yes	Yes	26.12	11.53	1.73	8.74

Table 3: Effect of CTC on Fbk-to-DSU PT and/or ST FT to the DSU-Adapter method. All results are in BLEU and the best in each group is denoted by '_'.

different training stages. Owing to the large number of vocabulary configurations on Table 2, we only experiment with: 1) "No-BPE 1K-8K", 2) "BPE 8K", 3) "BPE 32K" and 4) "BPE 32K-8K". In each training stage, we report the effect of CTC to the ST performance (per resource group) by averaging the BLEU of these 4 configurations.

Table 3 presents the analysis of applying CTC on our DSU-Adapter method. The training condition "Has CTC in DSU PT" refers to the case of applying CTC on the *discrete speech units* in Fbk-to-DSU pretraining, whereas "Has CTC in ST FT" refers to the case of applying CTC on the *translations* in ST finetuning, i.e., on the paired Fbk-Trl data. Our result shows that CTC helps on either stage, but the gain is 0.27 BLEU more in ST finetuning. Using them jointly still helps, but the marginal gain is barely 0.03 BLEU.

6 Limitations and future works

In the previous sections, we discuss the noticeable benefits of our DSU-pretraining method in creating a more compact ST model. In spite of this, there are several factors that are not thoroughly explored and could improve the model performance further:

K-Means clustering We did not inspect the clustering size (fixed to 1,000) and the number of training instances (only fixed to 98,000) used in training the K-Means clustering model. Apart from tuning its hyper-parameters, using other techniques, such as residual vector quantisation (Zeghidour et al., 2021; Défossez et al., 2022) and multiple codebooks (Guo et al., 2023), might bring better improvement.

Other acoustic encoders We did not experiment other acoustic encoders, such as conformer (Gulati et al., 2020; Papi et al., 2023) and E-Branchformer (Peng et al., 2023). This stronger encoders should provide further gains for our method since they also enjoy the benefit of pretraining.

A stronger pretrained decoder Apart from strengthening the encoder, the DSU-to-Trl model and hence its decoder (used in initialisation) could also be improved, e.g. via back-translation, upsampling the textual sequence (Yan et al., 2024) and pretraining with more text data, while maintaining the small decoder size.

Further analyses In addition to improving our pretraining method for better model compactness, there are other related research directions worth further analyzing. One direction would be how, in terms of acoustic pretraining, DSU compared with transcripts (if available in that language) over different data scales. Another interesting research direction would be the comparison and analysis of using DSU or dense features in a large pretrained model setting, such as Whisper (Radford et al., 2023) and Large Language Models.

7 Conclusion

In this paper, we consider a memory-constrained setting for ST. Our proposed method uses DSU in the form of pretraining to distil the knowledge from the Self-Supervised Speech model to creating more compact Speech-to-text Translation. Our compact model, i.e., the DSU-Adapter, shows strong and consistent improvements in three evaluation metrics over the baselines. In contrast to using DSU as model inputs, our method does not require quantization modules in inference and shows stronger robustness to the DSU tokenization. Finally, our method requires no transcripts, making it also suitable for low-resource setting.

Acknowledgements

This work was funded by UK Research and Innovation (UKRI) under the UK government's Horizon Europe funding guarantee (grant number 10039436: UTTER). The computations described in this research were performed using the Baskerville Tier 2 HPC service (https://www.baskerville.ac.uk/). Baskerville was funded by the EPSRC and UKRI through the World Class Labs scheme (EP/T022221/1) and the Digital Research Infrastructure programme (EP/W032244/1) and is operated by Advanced Research Computing at the University of Birmingham.

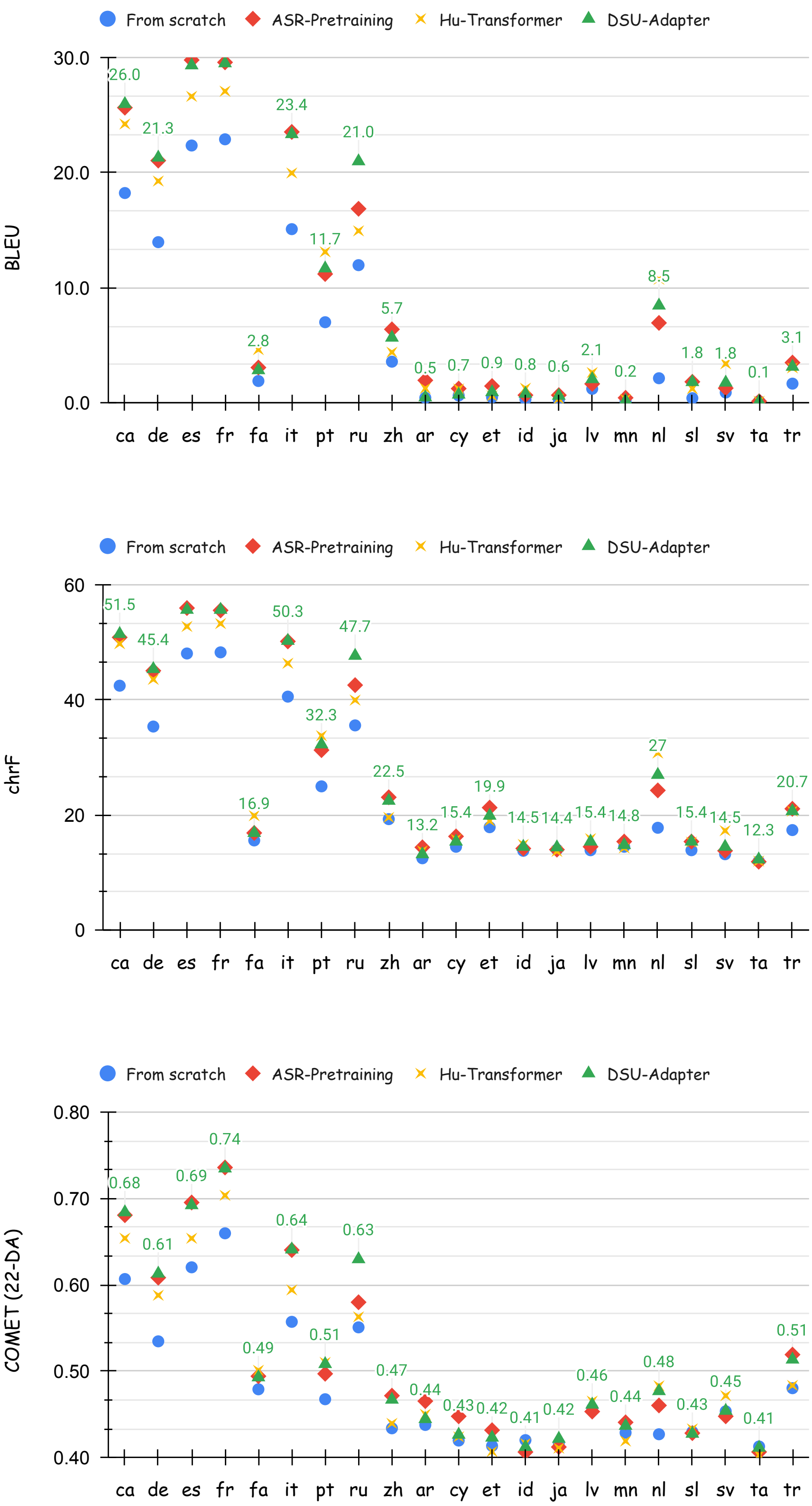

Figure 3: Results in BLEU, chrF and COMET-22-DA on each language pair of CoVoST-2 (X-En).

References

Milind Agarwal, Sweta Agrawal, Antonios Anastasopoulos, Luisa Bentivogli, Ondřej Bojar, Claudia Borg, Marine Carpuat, Roldano Cattoni, Mauro Cettolo, Mingda Chen, William Chen, Khalid Choukri, Alexandra Chronopoulou, Anna Currey, Thierry Declerck, Qianqian Dong, Kevin Duh, Yannick Estève, Marcello Federico, Souhir Gahbiche, Barry Haddow, Benjamin Hsu, Phu Mon Htut, Hirofumi Inaguma, Dávid Javorský, John Judge, Yasumasa Kano, Tom Ko, Rishu Kumar, Pengwei Li, Xutai Ma, Prashant Mathur, Evgeny Matusov, Paul McNamee, John P. McCrae, Kenton Murray, Maria Nadejde, Satoshi Nakamura, Matteo Negri, Ha Nguyen, Jan Niehues, Xing Niu, Atul Kr. Ojha, John E. Ortega, Proyag Pal, Juan Pino, Lonneke van der Plas, Peter Polák, Elijah Rippeth, Elizabeth Salesky, Jiatong Shi, Matthias Sperber, Sebastian Stüker, Katsuhito Sudoh, Yun Tang, Brian Thompson, Kevin Tran, Marco Turchi, Alex Waibel, Mingxuan Wang, Shinji Watanabe, and Rodolfo Zevallos. 2023. FINDINGS OF THE IWSLT 2023 EVALUATION CAMPAIGN. In *Proceedings of the 20th International Conference on Spoken Language Translation (IWSLT 2023)*, pages 1–61, Toronto, Canada (in-person and online). Association for Computational Linguistics.

Junyi Ao, Ziqiang Zhang, Long Zhou, Shujie Liu, Haizhou Li, Tom Ko, Lirong Dai, Jinyu Li, Yao Qian, and Furu Wei. 2022. Pre-Training Transformer Decoder for End-to-End ASR Model with Unpaired Speech Data. In *Proc. Interspeech 2022*, pages 2658–2662.

Arun Babu, Changhan Wang, Andros Tjandra, Kushal Lakhotia, Qiantong Xu, Naman Goyal, Kritika Singh, Patrick von Platen, Yatharth Saraf, Juan Pino, et al. 2021. Xls-r: Self-supervised cross-lingual speech representation learning at scale. *arXiv preprint arXiv:2111.09296*.

Alexei Baevski, Yuhao Zhou, Abdelrahman Mohamed, and Michael Auli. 2020. wav2vec 2.0: A framework for self-supervised learning of speech representations. *Advances in neural information processing systems*, 33:12449–12460.

Xuankai Chang, Brian Yan, Kwanghee Choi, Jeeweon Jung, Yichen Lu, Soumi Maiti, Roshan Sharma, Jiatong Shi, Jinchuan Tian, Shinji Watanabe, et al. 2023. Exploring speech recognition, translation, and understanding with discrete speech units: A comparative study. *arXiv preprint arXiv:2309.15800*.

Sanyuan Chen, Chengyi Wang, Zhengyang Chen, Yu Wu, Shujie Liu, Zhuo Chen, Jinyu Li, Naoyuki Kanda, Takuya Yoshioka, Xiong Xiao, et al. 2022. Wavlm: Large-scale self-supervised pre-training for full stack speech processing. *IEEE Journal of Selected Topics in Signal Processing*, 16(6):1505–1518.

Ju-Chieh Chou, Chung-Ming Chien, Wei-Ning Hsu, Karen Livescu, Arun Babu, Alexis Conneau, Alexei Baevski, and Michael Auli. 2023. Toward joint language modeling for speech units and text. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 6582–6593, Singapore. Association for Computational Linguistics.

Alexandre Défossez, Jade Copet, Gabriel Synnaeve, and Yossi Adi. 2022. High fidelity neural audio compression. *arXiv preprint arXiv:2210.13438*.

Qingkai Fang and Yang Feng. 2023. Back translation for speech-to-text translation without transcripts. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4567–4587, Toronto, Canada. Association for Computational Linguistics.

Marco Gaido, Mauro Cettolo, Matteo Negri, and Marco Turchi. 2021. CTC-based compression for direct speech translation. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 690–696, Online. Association for Computational Linguistics.

Alex Graves, Santiago Fernández, Faustino J. Gomez, and Jürgen Schmidhuber. 2006. Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks. In *Machine Learning, Proceedings of the Twenty-Third International Conference (ICML 2006), Pittsburgh, Pennsylvania, USA, June 25-29, 2006*, volume 148 of *ACM International Conference Proceeding Series*, pages 369–376. ACM.

Anmol Gulati, James Qin, Chung-Cheng Chiu, Niki Parmar, Yu Zhang, Jiahui Yu, Wei Han, Shibo Wang, Zhengdong Zhang, Yonghui Wu, and Ruoming Pang. 2020. Conformer: Convolution-augmented Transformer for Speech Recognition. In *Proc. Interspeech 2020*, pages 5036–5040.

Liyong Guo, Xiaoyu Yang, Quandong Wang, Yuxiang Kong, Zengwei Yao, Fan Cui, Fangjun Kuang, Wei Kang, Long Lin, Mingshuang Luo, et al. 2023. Predicting multi-codebook vector quantization indexes for knowledge distillation. In *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5. IEEE.

Wei-Ning Hsu, Benjamin Bolte, Yao-Hung Hubert Tsai, Kushal Lakhotia, Ruslan Salakhutdinov, and Abdelrahman Mohamed. 2021. Hubert: Self-supervised speech representation learning by masked prediction of hidden units. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 29:3451–3460.

Jacob Kahn, Morgane Rivière, Weiyi Zheng, Evgeny Kharitonov, Qiantong Xu, Pierre-Emmanuel Mazaré, Julien Karadayi, Vitaliy Liptchinsky, Ronan Collobert, Christian Fuegen, Tatiana Likhomanenko, Gabriel Synnaeve, Armand Joulin, Abdelrahman Mohamed, and Emmanuel Dupoux. 2020. Libri-light: A benchmark for ASR with limited or no supervision. In *2020 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2020,* *Barcelona, Spain, May 4-8, 2020*, pages 7669–7673. IEEE.

Taku Kudo and John Richardson. 2018. SentencePiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 66–71, Brussels, Belgium. Association for Computational Linguistics.

Kushal Lakhotia, Eugene Kharitonov, Wei-Ning Hsu, Yossi Adi, Adam Polyak, Benjamin Bolte, Tu-Anh Nguyen, Jade Copet, Alexei Baevski, Abdelrahman Mohamed, and Emmanuel Dupoux. 2021. On generative spoken language modeling from raw audio. *Transactions of the Association for Computational Linguistics*, 9:1336–1354.

Phuong-Hang Le, Hongyu Gong, Changhan Wang, Juan Pino, Benjamin Lecouteux, and Didier Schwab. 2023. Pre-training for speech translation: CTC meets optimal transport. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pages 18667–18685. PMLR.

Yuchen Liu, Junnan Zhu, Jiajun Zhang, and Chengqing Zong. 2020. Bridging the modality gap for speech-to-text translation. *arXiv preprint arXiv:2010.14920*.

Toan Q. Nguyen and Julian Salazar. 2019. Transformers without tears: Improving the normalization of self-attention. In *Proceedings of the 16th International Conference on Spoken Language Translation*, Hong Kong. Association for Computational Linguistics.

Myle Ott, Sergey Edunov, Alexei Baevski, Angela Fan, Sam Gross, Nathan Ng, David Grangier, and Michael Auli. 2019. fairseq: A fast, extensible toolkit for sequence modeling. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics (Demonstrations)*, pages 48–53, Minneapolis, Minnesota. Association for Computational Linguistics.

Sara Papi, Marco Gaido, Andrea Pilzer, and Matteo Negri. 2023. When good and reproducible results are a giant with feet of clay: The importance of software quality in nlp. *arXiv preprint arXiv:2303.16166*.

Daniel S. Park, William Chan, Yu Zhang, Chung-Cheng Chiu, Barret Zoph, Ekin D. Cubuk, and Quoc V. Le. 2019. SpecAugment: A Simple Data Augmentation Method for Automatic Speech Recognition. In *Proc. Interspeech 2019*, pages 2613–2617.

Ankita Pasad, Bowen Shi, and Karen Livescu. 2023. Comparative layer-wise analysis of self-supervised speech models. In *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5. IEEE.

Yifan Peng, Kwangyoun Kim, Felix Wu, Brian Yan, Siddhant Arora, William Chen, Jiyang Tang, Suwon Shon, Prashant Sridhar, and Shinji Watanabe. 2023. A Comparative Study on E-Branchformer vs Conformer in Speech Recognition, Translation, and Understanding Tasks. In *Proc. INTERSPEECH 2023*, pages 2208–2212.

Adam Polyak, Yossi Adi, Jade Copet, Eugene Kharitonov, Kushal Lakhotia, Wei-Ning Hsu, Abdelrahman Mohamed, and Emmanuel Dupoux. 2021. Speech Resynthesis from Discrete Disentangled Self-Supervised Representations. In *Proc. Interspeech 2021*, pages 3615–3619.

Matt Post. 2018. A call for clarity in reporting BLEU scores. In *Proceedings of the Third Conference on Machine Translation: Research Papers*, pages 186–191, Brussels, Belgium. Association for Computational Linguistics.

Alec Radford, Jong Wook Kim, Tao Xu, Greg Brockman, Christine McLeavey, and Ilya Sutskever. 2023. Robust speech recognition via large-scale weak supervision. In *International Conference on Machine Learning*, pages 28492–28518. PMLR.

Ricardo Rei, José G. C. de Souza, Duarte Alves, Chrysoula Zerva, Ana C Farinha, Taisiya Glushkova, Alon Lavie, Luisa Coheur, and André F. T. Martins. 2022. COMET-22: Unbabel-IST 2022 submission for the metrics shared task. In *Proceedings of the Seventh Conference on Machine Translation (WMT)*, pages 578–585, Abu Dhabi, United Arab Emirates (Hybrid). Association for Computational Linguistics.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Neural machine translation of rare words with subword units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1715–1725, Berlin, Germany. Association for Computational Linguistics.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems*, 30.

Changhan Wang, Yun Tang, Xutai Ma, Anne Wu, Dmytro Okhonko, and Juan Pino. 2020. Fairseq S2T: Fast speech-to-text modeling with fairseq. In *Proceedings of the 1st Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 10th International Joint Conference on Natural Language Processing: System Demonstrations*, pages 33–39, Suzhou, China. Association for Computational Linguistics.

Changhan Wang, Anne Wu, Jiatao Gu, and Juan Pino. 2021. CoVoST 2 and Massively Multilingual Speech Translation. In *Proc. Interspeech 2021*, pages 2247–2251.

Felix Wu, Kwangyoun Kim, Shinji Watanabe, Kyu Jeong Han, Ryan McDonald, Kilian Q. Weinberger, and Yoav Artzi. 2023. Wav2seq: Pre-training speech-to-text encoder-decoder models using pseudo languages. In *IEEE International Conference on Acoustics, Speech and Signal Processing ICASSP 2023, Rhodes Island, Greece, June 4-10, 2023*, pages 1–5. IEEE.

Brian Yan, Xuankai Chang, Antonios Anastasopoulos, Yuya Fujita, and Shinji Watanabe. 2024. Cross-modal multi-tasking for speech-to-text translation via hard parameter sharing. In *ICASSP 2024 - 2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 11941–11945.

Neil Zeghidour, Alejandro Luebs, Ahmed Omran, Jan Skoglund, and Marco Tagliasacchi. 2021. Soundstream: An end-to-end neural audio codec. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 30:495–507.

Biao Zhang, Barry Haddow, and Rico Sennrich. 2022a. Revisiting end-to-end speech-to-text translation from scratch. In *International Conference on Machine Learning*, pages 26193–26205. PMLR.

Biao Zhang, Barry Haddow, and Rico Sennrich. 2023a. Efficient CTC regularization via coarse labels for end-to-end speech translation. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 2264–2276, Dubrovnik, Croatia. Association for Computational Linguistics.

Dong Zhang, Rong Ye, Tom Ko, Mingxuan Wang, and Yaqian Zhou. 2023b. DUB: Discrete unit back-translation for speech translation. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 7147–7164, Toronto, Canada. Association for Computational Linguistics.

Ziqiang Zhang, Long Zhou, Junyi Ao, Shujie Liu, Lirong Dai, Jinyu Li, and Furu Wei. 2022b. SpeechUT: Bridging speech and text with hidden-unit for encoder-decoder based speech-text pre-training. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 1663–1676, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.